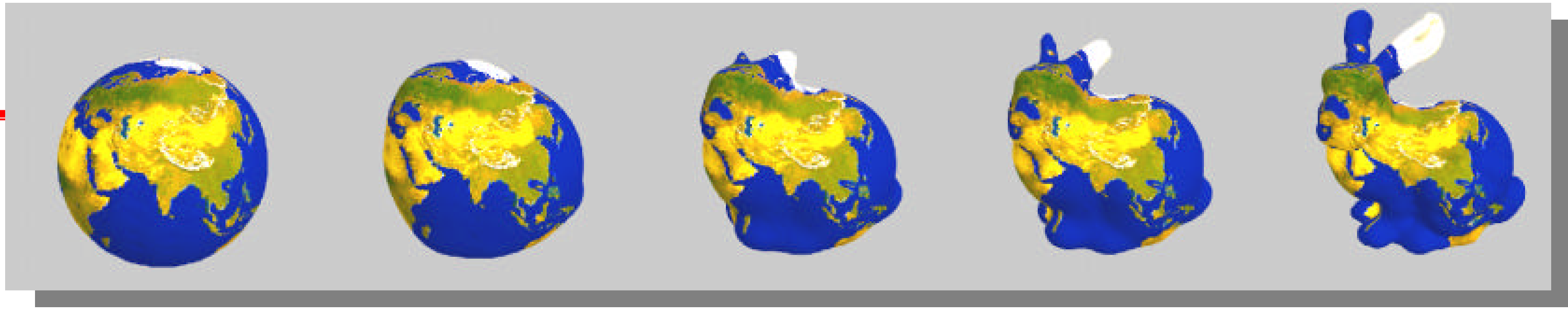


NSF CARGO: Multi-scale Topological Analysis of Deforming Shapes

*APES (**A**nalysis and **P**arameterization of **E**volving **S**hapes)*

Jarek Rossignac

Andrzej Szymczak

Greg Turk

College of Computing

Georgia Tech, Atlanta

Rob Ghrist

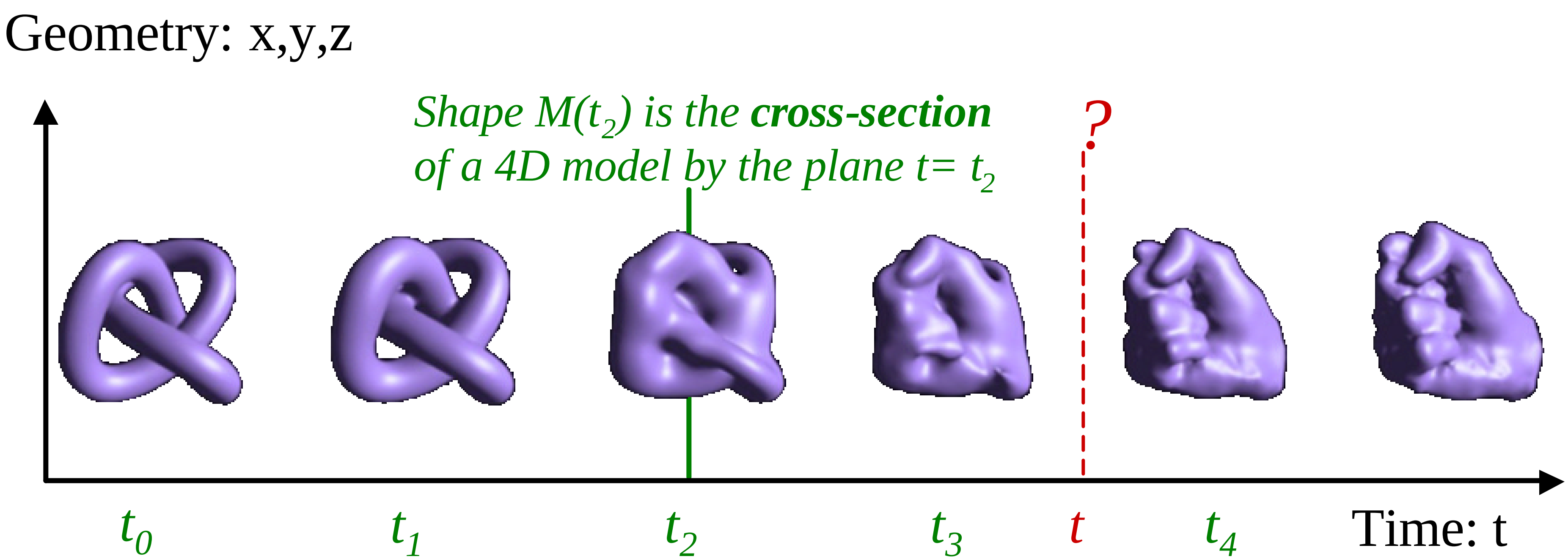
School of Mathematics

Georgia Tech, Atlanta

A 4D model of the behavior of 3D shapes

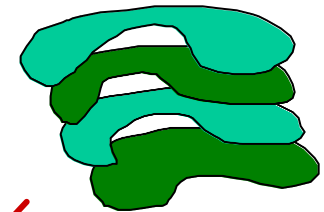
Many animation and simulation packages represent behavior as a series of independent **3D frames**

Yet, a continuous model is better suited for supporting slow-motion, geometric and topological **analysis**, and coherent **segmentation**, **texturing** and **visualization**

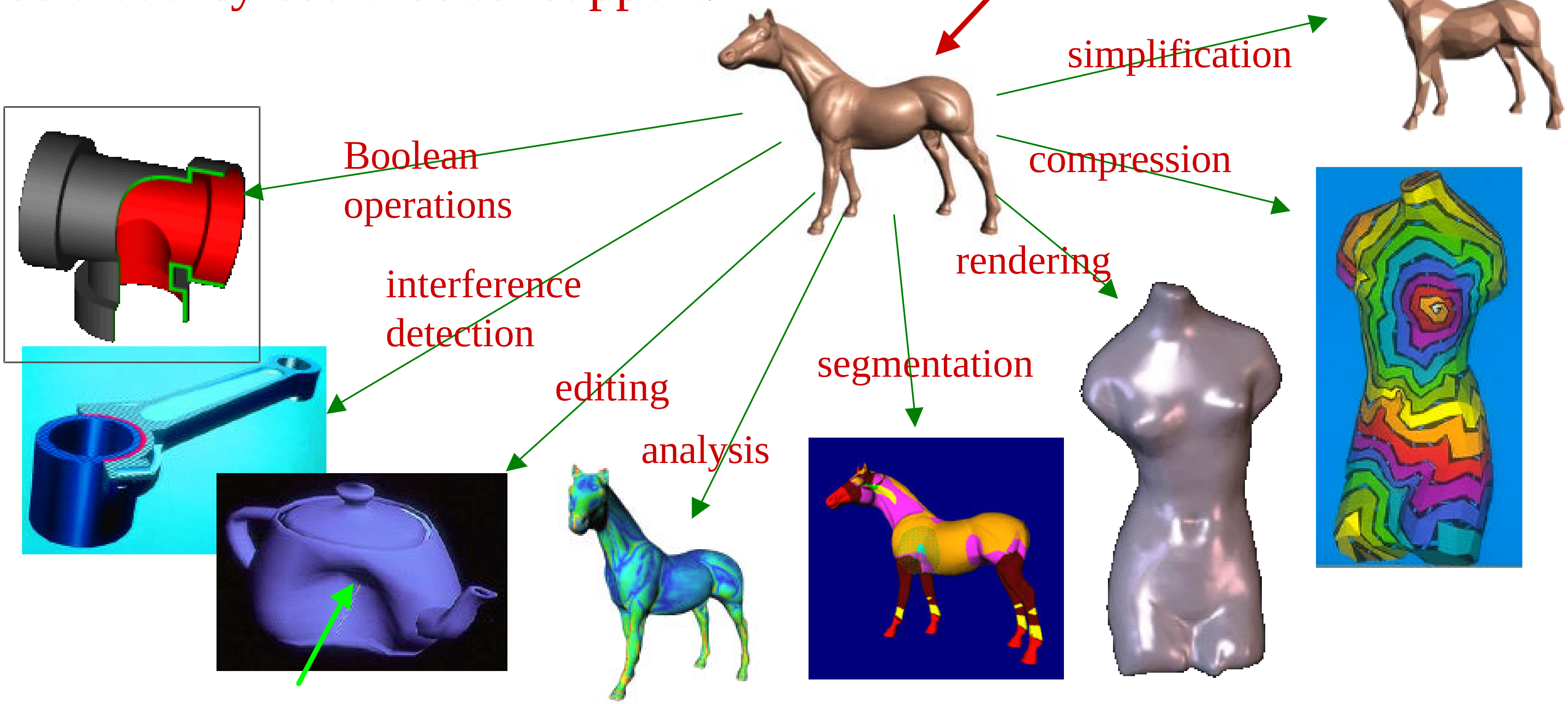


3D applications migrated from slices to 3D

Early 3D GIS, Medical, and M-CAD systems were based on stacks of **cross-sections**

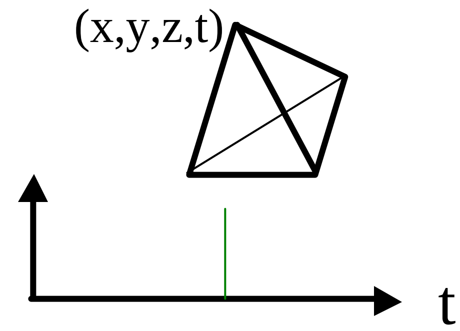


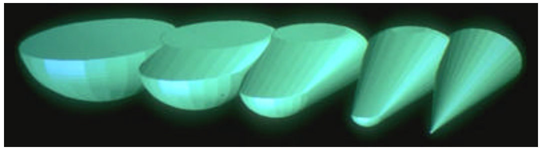
They migrated to **continuous** 3D models, so that they could better support:

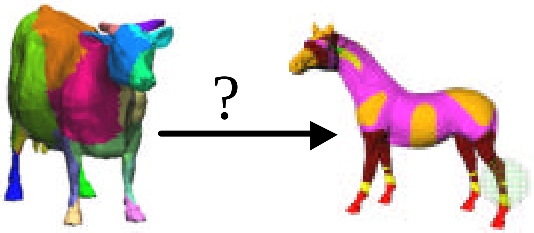


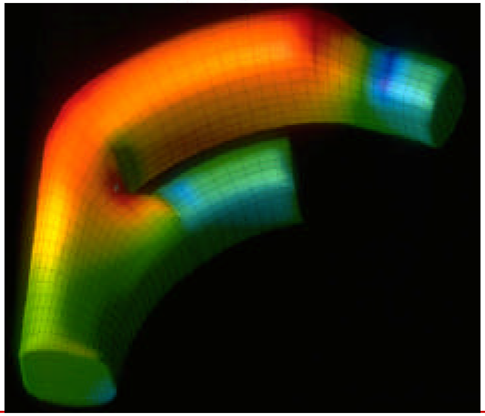
Need a similar migration for animations

- **Represent & slice** a **hyper-surface** in 4D
  - Voxels or Tetrahedra in 4D:
    - (x,y,z,t)+connectivity?
  - Fast slice of hypercubes or tetrahedra
    - Addressed by Jack Snoeyink’s CARGO project
- Generate **interpolating** 4D models
  - 3D morph, fitting implicit hyper-surface
- Use 4D model to build temporally **coherent** **<u>segmentations</u>** of the evolving shape into **features**?
- Use 4D model to build temporally **coherent** **<u>parameterizations</u>** of the evolving **features**?









How to generate a 4D model?

- **Design by manipulating control points of B-spline S(u,v,t)**
- **Fit a hyper-surface to constraints (discussed by Greg Turk)**
- **Piecewise linear or polynomial morphs between 3D frames**
- **…**

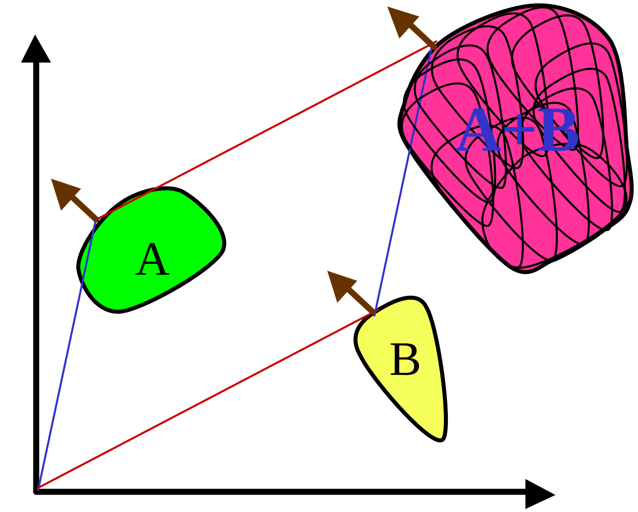
3D morphing via Minkowski averaging

- **A+B = {a+b: a î A, b î B}**
  - Matches boundary points with same normal
- **M(t)=(1-t)A+tB**

"Solid-Interpolating Deformations: Construction and Animation of PIPs",

Kaul&Rossignac, C&G'92, 16(1)107-115.

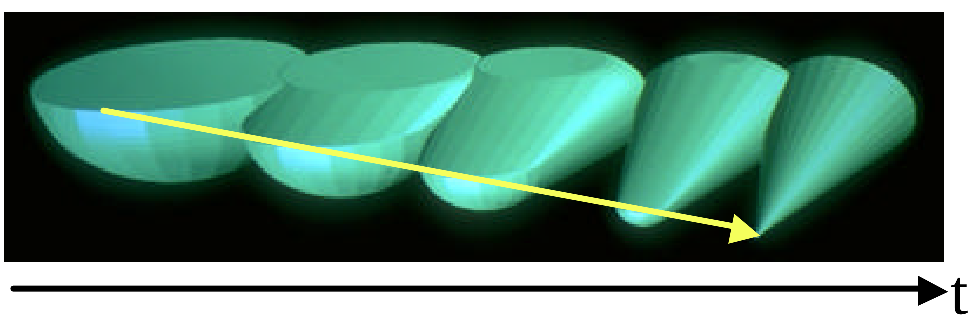
  - Constant connectivity, linear trajectory
  - Realtime animation

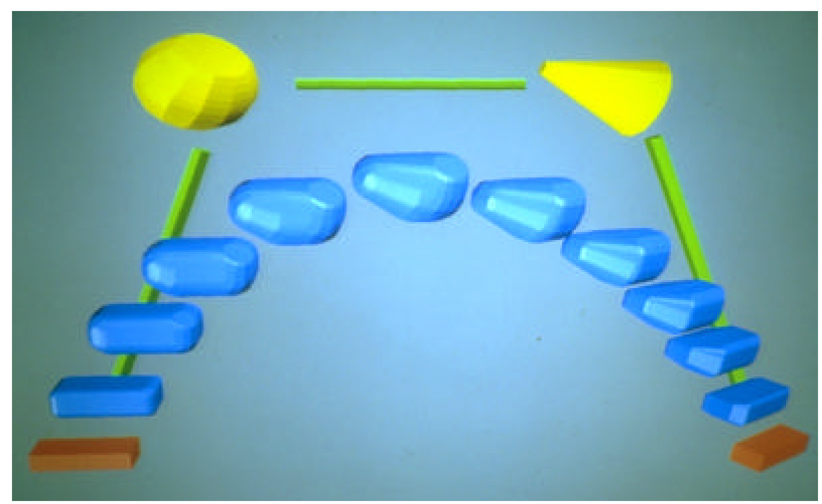


M(t)= (1-t)((1-t)((1-t)A+tB)+t((1-t)B+tC))+t((1-t)((1-t)B+tC)+t((1-t)C+tD))

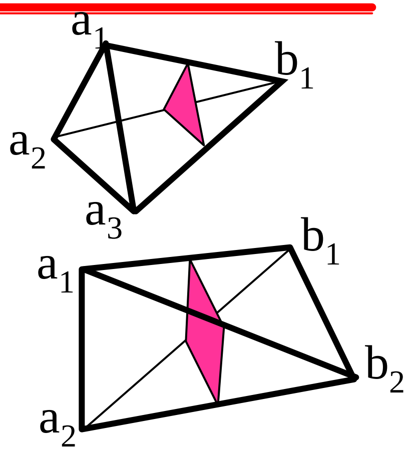
"AGRELs and BIPs: Metamorphosis as a Bezier curve in the space of polyhedra", Rossignac&Kaul, CGForum'94, 13(3)179-184.

  - Vertices move on Bezier curves

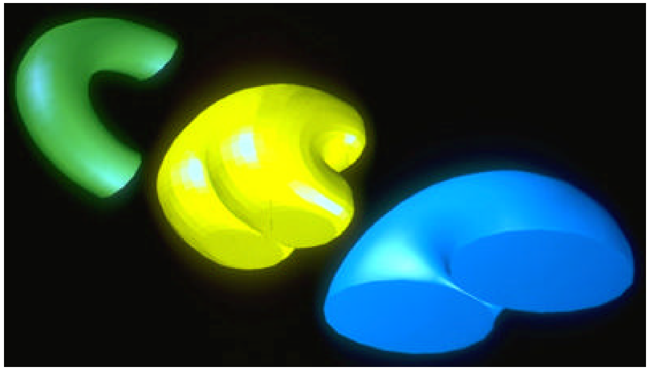


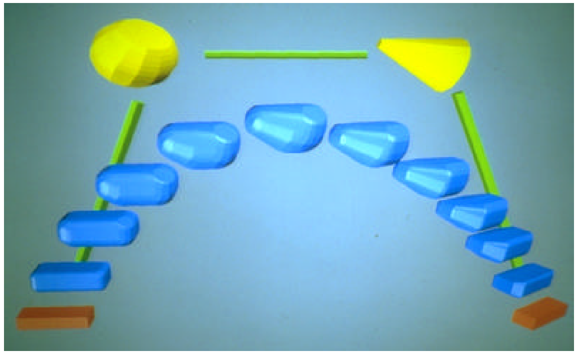


From 3D morphs to tets (tetrahedra) in 4D



- **Each vertex of M(t)=(1-t)A+tB**
  - linearly interpolates a vertex of A and a vertex of B
- **The faces of M(t) are time slices of tets in 4D**
  - 1, 2, or 3 vertices of a tet are on A
- **Tets establish mapping**
  - Vertex-triangle
  - Edge-edge
  - Triangle-vertex
- **Research: Non-convex cases**
  - Pairwise disjoint tets
  - Minimal total distance or volume?
- **Research: Temporal coherence**
  - Smoothness and key-frame interpolation



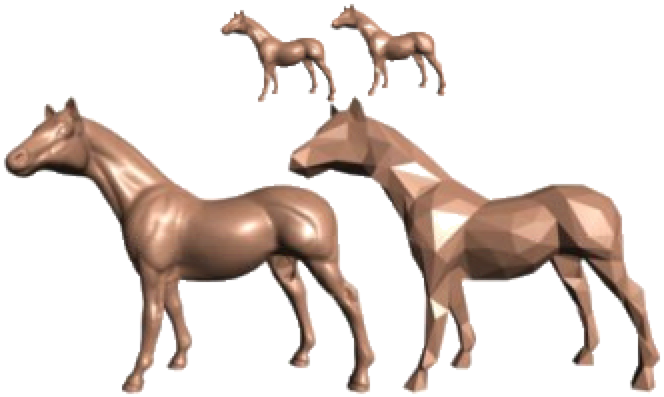


Extending analysis/segmentation to 4D

- **Segment each 3D frame independently and try making them coherent**
- **Segment the 4D model**
- **Want multi-resolution to ignore high frequency details**

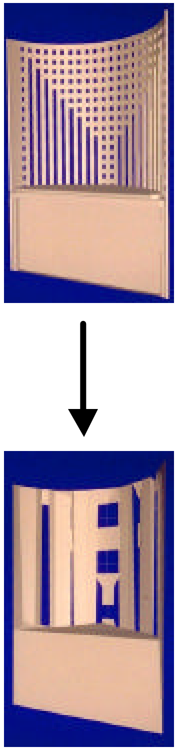
May need a simplified 4D model

- **A detailed (tet) model that interpolates all slices may be too detailed for rapid transmission or animation**
- **It may not be suited for analyzing its gross features**

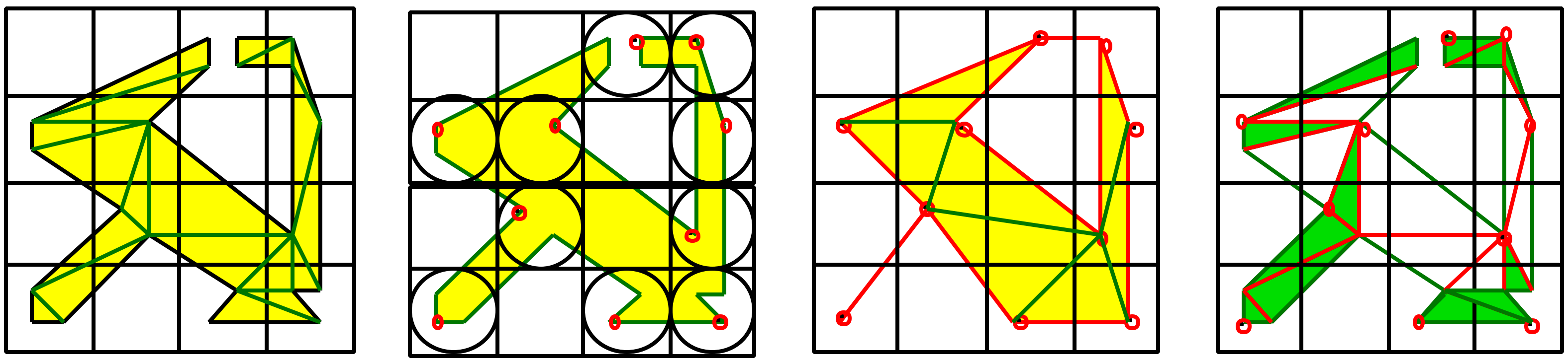


- **We want less-detailed approximations**
  - For transmission of Levels-of-Detail
  - To accelerate animation
  - For multi-resolution analysis of animations
- **We propose extend simplification techniques developed for meshes in 3D to tetrahedral meshes in 4D**
  - Better coherence than simplifying each 3D frame independently
    - May for example simplify a **short** appearance of a protrusion

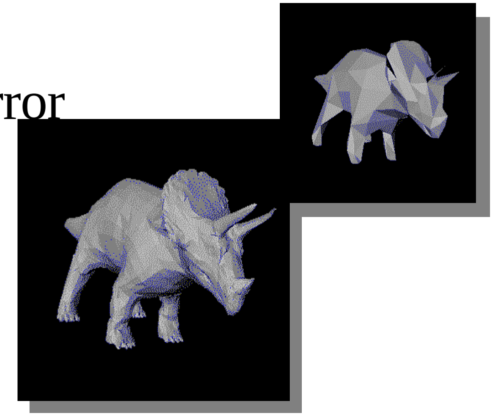
3D simplification techniques (LOD)

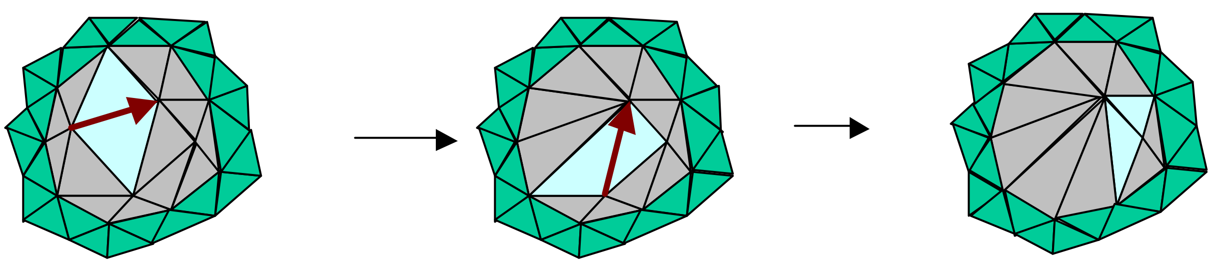


- **Quantize & cluster vertex data (Rossignac&Borrel'92)**
  - remove degenerate triangles (that have coincident vertices)



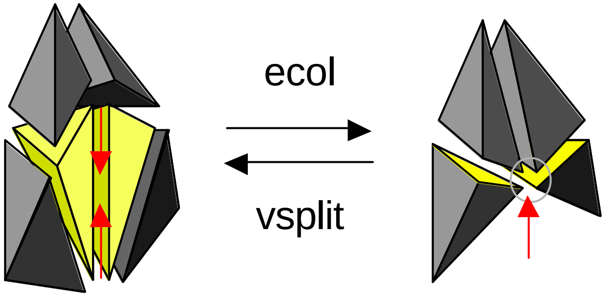
  - Adapted by Lindstrom for out-of-core simplification
- **Repeatedly collapse best edge (Ronfard&Rossignac96)**
  - while minimizing bound on **maximum** error
  - Adapted by M. Garland for **mean square** (quadric) error



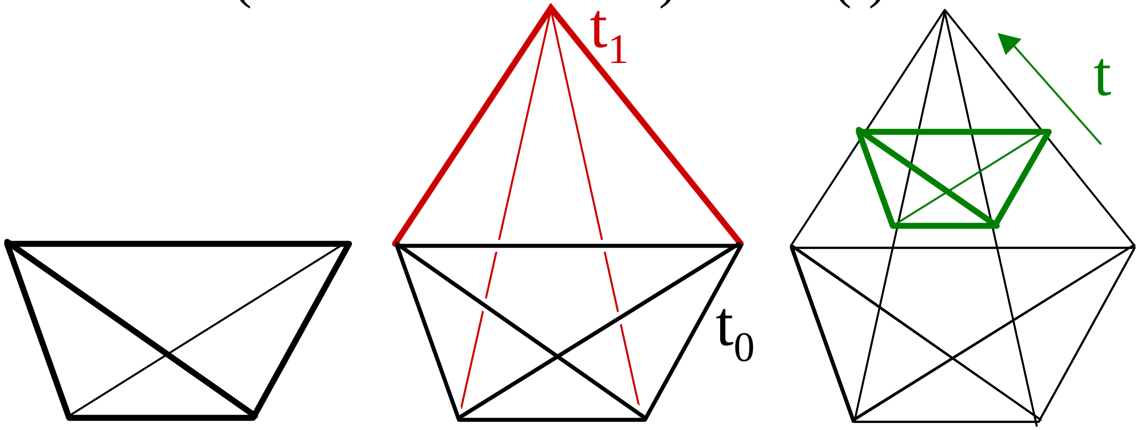


4D extensions of 3D simplifications

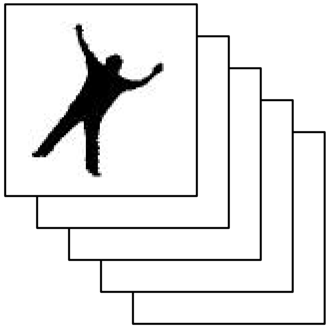
- **Edge-collapses were extended to tetrahedral meshes in 3D**
  "Implant Sprays: Compression of Progressive Tetrahedral Mesh Connectivity", Pajarola, Rossignac, and Szymczak, IEEE Visualization 1999.



- **Need a 4D error estimator (isotropic?)**
- **Get a continuous family of 4D models**
  - Each vertex at one level of detail linearly evolves towards its representative in the cruder model (Geomorph)
  - Each evolving tetrahedron is a (constant-resolution) slice T(r) of a pentatope in 5D



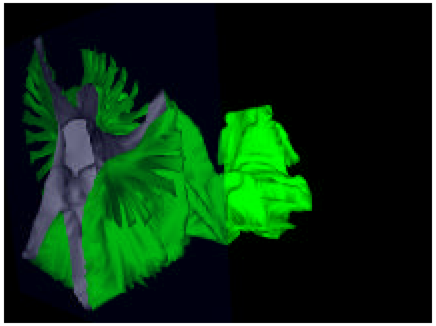
2D experiment: Multiresolution cel animation



Stack of 2D slices

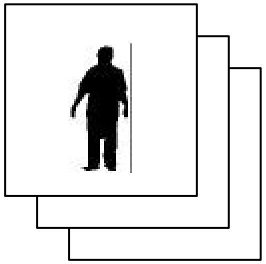
Fit surface in 3D

Simplify surface



Cross-section

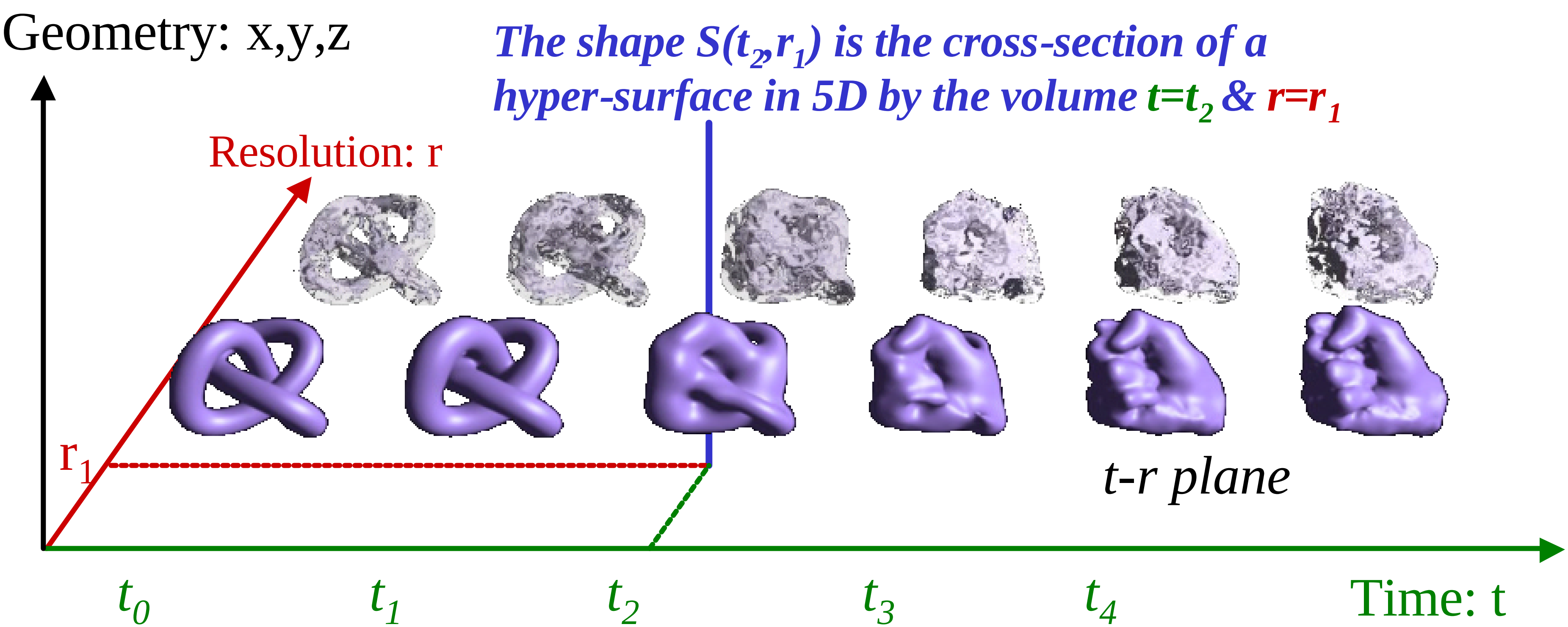
"Surface simplification and Edgebreaker compression for 2D Cel Animations", Kwatra & Rossignac, Shape Modeling International, 2001.



Simplified animation

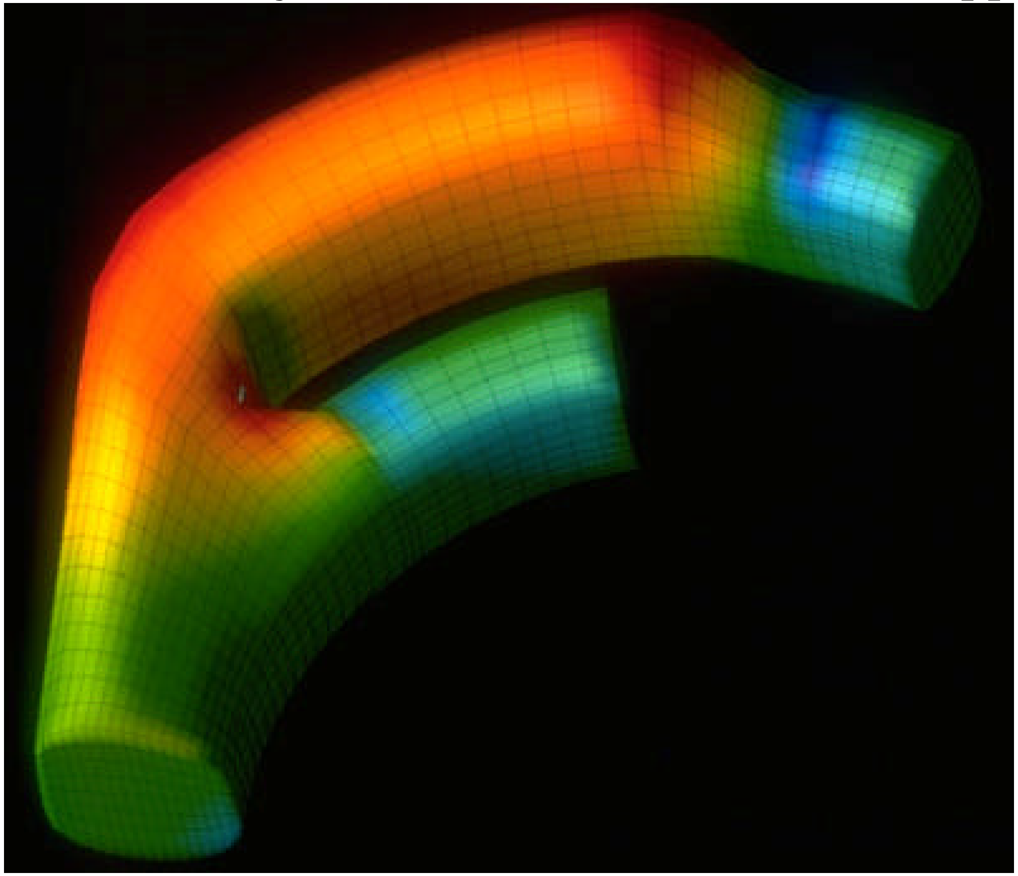
A 5D multi-resolution behavior model

- **We want to create a continuous family S(t,r) of 3D models parameterized by time t and resolution r**
- **We will represent it as a hyper-surface in 5D: A penta-mesh**



Segmentation and parameterization

- **Want segmentation and parameterization of S(t,r) that is coherent with respect to t and r.**
  - For multi-resolution behavior analysis and for coherent texture mapping

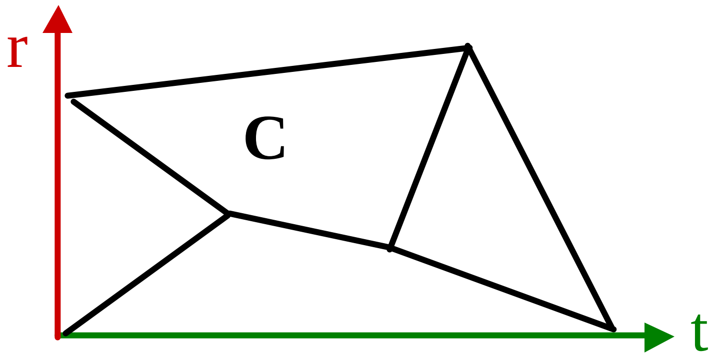


APES Objective

- **Build a multi-resolution model of evolving 3D shapes**
  - Consider a **time-dependent** family of "surfaces" M(t) and a process producing a "simplification" S(M(t),r), for simplicity denoted **S(t,r)**, that **approximates** M within a given "resolution" r.
    - As t and/or r evolve, the shape and **topology** of S(t,r) may change.
- **Infer a coherent segmentation and parameterization**
  - A **segmentation** of S(t,r) into "natural features" coherent as t or r evolve
    - Some features may appear or disappear as t and r evolve
  - A **parameterization** $F_{(t,r)}(u,v)$ of each feature F that changes "smoothly" with t and r
    - Will support texturing and analysis of evolution
  - A **decomposition** of the **t-r plane** into **cells**, such that within a given cell, C, the topology of S(t,r), its segmentation into features and the connectivity of these features remains **constant**.
    - *The precise definitions of the "vague" terms will evolve as we match application needs against theoretical and practical limitations.*

Given S(t,r), APES will build

- A **decomposition** of the t-r plane into cells and the association with each cell of the list of its active features.
- A continuous 1-to-1 **map** C(t,r,F,u,v), from some generic domain in t-r-u-v space to the surface of a feature F, which given a point (t,r) in cell C, a feature-Id F, and two parameters (u,v) will return a point on S(t,r).
- A mapping (**junction chart**) from (F,u,v) to (F',u',v') which encodes the conversion between the two parameterizations at the common boundary of two adjacent cells.



Theory, data-structure, algorithms for

- **Representing** the evolution model M and its multi-resolution version S
- Computing M through **interpolation** of 3D frames or 4D samples
- Computing S through **simplification** of M
- **Segmenting** S(t,r) into topologically simple and domain dependent features
- Identifying where the **topology** of S(t,r) **changes**
- **Parameterizing** the features on individual frames, on M, and on S
- **Aligning** the parameterization to the natural orientation of features
- **Slicing** each feature to texture and render it in the desired (t,r) section
- **Decomposing** the t-r plane into cells of constant features and topology
- Supporting **conversion** between parameterizations in adjacent cells
- Measuring and categorizing shape evolution at different resolutions